Journal of Library Automation
Vol 4/4 Dec. 1971

AUTOMATIC PROCESSING OF PERSONAL NAMES FOR FILING

Foster M. PALMER: Associate University Librarian, Harvard University Library, Cambridge, Massachusetts

Describes a method for preparing personal names already in machine readable form for processing by any standard computer sort program, determining filing order insofar as possible from normally available information rather than from special formating. Prefix recognition is emphasized; multiword forename entries are a problem area. Provision is made for an edit list of problems requiring human decision. Possible extension of the method to titles is discussed.

This paper describes a method of computerized filing of personal names for display in book catalogs or other lists intended for direct human consultation. The problem is to be distinguished from a related but different one: computerized storage for retrieval by means of a search key, in which machine rather than human convenience can determine the order.

To the extent that filing is a purely mechanistic sorting process, it is ideally suited to computerization. However, it was early recognized that there are many possible complications in machine filing of library entries, even in the relatively straightforward area of personal names. Some of these complications arise from such factors as upper-case codes, diacritic codes, and punctuation; others are the result of library rules or practices that call for departures from strict alphabetical order. While the latter are especially numerous in subject headings and titles, they affect names as well, for example, the custom of filing Mc as if Mac.

While no general review of the literature on machine filing will be attempted here, attention will be called to selected contributions. Nugent (1) described an approach to computerizing the Library of Congress filing rules and pointed out areas where the present rules do not lend themselves to mechanization. Cartwright and Shoffner (2) discussed four major ways of approaching a solution to the problem and concluded that a mixture of different methods would eventually be required. In a later publication Cartwright (3) developed his ideas further and included a brief description of the present writer's then unpublished work. The principal monograph on the subject is that by Hines and Harris (4). They present a suggested filing code departing significantly from those in widespread use and propose that material be encoded in a certain fashion so that it will be ready for computer sorting. In particular, considerable dependence is placed on distinctions between single, double, and multiple blanks separating words or fields. In a recent paper, Harris and Hines restate their rules briefly and report on their later research (5).

The present paper describes a different, virtually an opposite, approach. Rather than relying on special formating of the material at the time of encoding, the system described herein attempts to derive the necessary filing information from normally formated material. Historically, it grew out of a desire to construct improved indexes for use at the Harvard University Library to the body of records distributed by the MARC Pilot Project, in which there were field indicators and a limited number of delimiters within fields, but a general absence of information added expressly for the purpose of filing.

While some early work embraced both personal names and titles, it was soon apparent that names by themselves presented a considerable challenge, and further consideration of the even more difficult areas of titles, corporate entries, and subject entries was deferred. A few comments on the possible applicability of the general method to titles will be made later.

The concrete form which the work eventually took was an Autocoder macro instruction for a second generation computer, an IBM 1401. (A macro instruction is a means of calling forth by means of a single instruction a more extensive routine already worked out and placed in the system "library.") Since the 1401 was a fairly small computer, it was important that the algorithm not require an excessive number of instructions, and since the internal speed of the machine was only moderate, it was also important that processing be direct and economical. The method used, however, is by no means limited to a particular computer or a particular language. A partial version of the algorithm has been written in ADPAC, as an exercise in the evaluation of that language, and run on an IBM 360-65 using MARC II test data.

The system is based on examination of names (previously identified as such by appropriate tags) and development of parallel sort keys consisting

only of letters, numerals, and blanks, readily processable by any standard computer sort package designed for alphanumeric information. The only requirements are that blank sort low and that the letters A - Z and the numerals 0 - 9 sort in their natural order; whether numbers are considered higher or lower than letters does not matter.

Processing starts at the beginning of the name and proceeds until one of three conditions prevails: The number of characters examined is equal to the length of the field as specified in the record; the number of characters developed in the sort key has reached a specified cut-off point or the default value of 40; or a delimiter indicating the end of the name, or the end of the name proper, is encountered (a search being then made beyond the delimiter for a date, which, if found, is added to the sort key).

The sort key is derived by transferring letters (or, in the case of a date, numbers) from the source, with occasional modifications as described below, and inserting one of four filing codes at the end of each word or element of the name. In early work, single special characters were used as filing codes, but this was inappropriate as a general solution since the filing order of these characters depended on the collating sequence peculiar to a particular computer. Furthermore, it was inconvenient because it involved changing all blanks to something else, since a blank within a name with its implication of something to follow should not file as low as whatever indicates the very end of the name. The idea of using a two-character code, the first always being blank so that any filing code will file ahead of any letter or date, was derived from Nugent (1) and has been followed in all later work. Only three filing codes were actually used in compiling indexes to the MARC I tapes, and in the first description privately circulated by the author (6). However, at least four are now seen to be necessary, actual need to distinguish the second and third not yet having been encountered but being possible:

Code (blank followed by:)	Placement
3	The end of the name including date if any.
5	Between the name proper and a date.
6	The end of the surname.
7	The end of any other "word" of the name. (A word is any element followed by a blank, hyphen, comma, or period, except that prefixes which are identified as such are not considered separate words.)

The following examples illustrate the use of the codes and the general workings of the system. In this and later examples, the left hand column gives data in MARC I format (where diacritics are represented by superscript numbers preceding the letters to which they apply, and the equal sign is a delimiter), and the right hand column gives the sort key as derived by the macro.

Arthur	arthur 3
Arthur, Joseph	arthur 6joseph 3
Arthur, Joseph,= 1875-	arthur 6joseph 51875 3
Arthur, Joseph Charles	arthur 6joseph 7charles 3
Arthur-Behenna, K.	arthur 7behenna 6k 3
Arthur-Petr²os, Gabriele Maria	arthur 7petros 6gabriele 7maria 3
Wilson, William	wilson 6william 3
Wilson, William,= 1923-	wilson 6william 51923 3
Wilson, William Lyne	wilson 6william 7lyne 3
Wilson-Browne, A. E.	wilson 7browne 6a 7e 3

The use of the numbers 3, 5, 6, and 7 is arbitrary to a degree. An interval was left between 3 and 5 so that the end of name code could be changed to 4 if the name were a subject rather than a main or added entry. No extra interval to accommodate added entry as distinguished from main entry was left because the author did not wish to encourage what he regards as an unwise practice. However, those who insist may easily substitute a new series of codes allowing for it.

The distinction between end of name and end of surname serves to bring simple forename entries, that is those consisting of a single word, e.g. Sophocles, ahead of similar surnames, e.g. Sophocles, Evangelinus Apostolides. No serious work has yet been undertaken on the problem of processing complex forenames, but the distinctive tagging of forenames in MARC II has made available a growing body of experimental data and the codes 1 (and 2 for subject) are reserved for possible future use in this connection, without any intent of prejudging the question whether complex forename entries should come before similar surnames. It is the view of the author that the filing of complex forename entries is one of the areas in which all librarians are on most uncertain grounds in assessing the preference and convenience of readers.

In handling such entries as Alexander, Mrs., or Maurice, Sister, the algorithm depends on the presence of a delimiter before Mrs. or Sister to avoid filing after Alexander, Milton or Maurice, Robert. Such delimiters were in fact present in the MARC Pilot Project data. Despite the limitations mentioned in dealing with multiple-word forename entries and with surnames lacking forenames, the algorithm is well suited to names in the normal modern pattern, namely a simple or compound surname followed by a comma and one or more given names or initials. Furthermore, very specifically, it deals with prefix names. Prefixes with apostrophes are taken care of by a general dropping out of apostrophes and other non-significant punctuation:

[L'Isle, Guillaume de]	lisle 6guillaume 7de 3
O'Brian, Robert Enlow	obrian 6robert 7enlow 3

The same feature also handles such names as the following:

Prud'homme, Louis Arthur	prudhomme 6louis 7arthur 3
Ta'Bois, Roland	tabois 6roland 3

Most prefixes, however, are dealt with by a specific search based on examining the first letter of each new "word" of the name. If the element begins with A, B, D, E, F, I, L, M, O, S, T, V, or Z, a branch is made to a prefix searching routine tailor-made for the particular letter. Taking names beginning with L as an example, if the second character is "e," "a," or "o," a prefix may be present; otherwise the prefix search is discontinued. If still searching and the third character is a blank or a hyphen, a prefix is adjudged to be present. The letters "le," "la," or "lo" are moved to the sort key output field. Three input and two output characters are counted, effectively skipping over the blank or hyphen. Similarly, if the third character is an "s" followed by a blank or a hyphen, "les," "los," or "las" is moved with a count of four input and three output. Otherwise there is no prefix.

La Place, Pierre Antoine de	laplace 6pierre 7antoine 7de 3
Las Cases, Philippe de	lascases 6philippe 7de 3
Le Fanu, Joseph Sheridan	lefanu 6joseph 7sheridan 3
Lo Presti, Salvatore	lopresti 6salvatore 3

Routines for other letters, similar in approach but varying in detail, produce similar results:

Degli Antoni, Carlo	degliantoni 6carlo 3
De La Roche, Mazo	delaroche 6mazo 3
Fitz Gibbon, Constantine	fitzgibbon 6constantine 3
Van der Bijl, Hendrick Johannes	vanderbijl 6hendrick 7johannes 3

The search for prefixes and quasi-prefixes is not limited to the first surname. It is and quite plainly should be extended to given names:

Bundy, McGeorge	bundy 6macgeorge 3
Bundy, Mary Lee	bundy 6mary 7lee 3

Whether it should be extended to later elements of compound surnames is problematical. Bowing to the fact that filing is as much an art as a science, in practice a compromise was reached: the prefix search was extended to compounds, except when the prefix of the succeeding element begins with D. The exception was made to accommodate the large number of Hispanic names in this pattern, since it seemed clearly preferable to file all the names beginning "Pérez de" before any of those beginning "Pérez del":

P²erez, Joaqu²in	perez 6joaquin 3
P²erez de Urbel, Justo	perez 7de 7urbel 6justo 3
P²erez del Castillo, Jos²e	perez 7del 7castillo 6jose 3
P²erez Gald²os, Benito	perez 7galdos 6benito 3

Perhaps skipping prefix treatment in subsequent elements should have been made the rule rather than the exception; but an exception would then have been required for "Mc," "St.," and perhaps others.

A list of the prefixes and quasi-prefixes sought for is given in Table 1. Note that in some cases the result is considered doubtful, and a special signal is set. In such situations the program can then set another signal within the macro and reprocess the name using alternate rules.

Table 1. List of Prefixes, Etc., Found by Special Search

A	1, 4, 7	Den		St.	4, 15
A	2, 4	Der	4, 11, 18	Ste.	16
Ab		Des		Te	4, 11
al	5	Di		Ten	4
Al	3, 4, 6	Do		Ter	
An	4, 7	Dos	4, 11	The	1, 4, 8
Ap		Du		Van	1, 17
At		el	5	Van	2, 4, 12, 17
Aus	17	El	3, 4, 6	Van'...	4, 9
Aus'...	4, 9	Fitz		Vande	
Bar	10	Im		Vanden	
Bat	10	In	17	Vander	
Ben	10	La		Ver	
Da		Las		Von	17
Das	4, 12	Le		Vonde	
De	17	Les		Vonden	
Degli	1	Lo		Vonder	
Dei		Los		Z	4, 5
Del		M'	4, 14	Zu	17
Della		Mac		Zum	
Delle		Mc	13	Zur	
Dello		O			

1. Only when followed by blank.
2. Only when followed by hyphen.
3. Only when upper case.
4. "Doubt" signal is set.
5. Bypassed, i.e. dropped out and disregarded.
6. Bypassed if "alternate" signal is on.
7. Bypassed unless "alternate" signal is on.
8. Bypassed if first word.
9. Aus'm and Van't are closed up to "ausm" and "vant" by the general dropping of apostrophes but no attempt is made at further special processing since their rarity would not justify the necessary elaboration of the algorithm.
10. Not treated as prefix if special parameter is present.
11. Not treated as prefix if "alternate" signal is on.
12. Not treated as prefix unless "alternate" signal is on.
13. Expanded to "mac".
14. Expanded to "mac" unless "alternate" signal is on.
15. Expanded to "saint".
16. Expanded to "sainte".
17. Another prefix may follow, as in De La.
18. Previous notes do not apply when preceded by Van or Von.

Diacritical marks on other than the first letter, or capitalization beyond the normal, such as all caps., would prevent proper processing. Except as indicated, lower case is included along with upper, and prefixes followed by a hyphen are treated the same as those followed by a blank. The MARC I corpus included several names with hyphenated prefixes, and fortuitously a method was available with the 1401 for giving the hyphen search almost a "free ride" along with that for the blank. Since the code for hyphen was a single bit, the so-called B bit, and a blank was represented by no bits, a "branch if bit equal" instruction specifying all the other bits, A, 8, 4, 2, and 1, would branch if any character other than blank or hyphen was present. Implementations for other machines may have to devote a disproportionate number of instructions to the search for the rare hyphenated prefixes, or else risk missing them.

No doubt some other prefixes could be added to the list. "Ua," for example, was considered but not included in the actual working macro after examination of a catalog of five million cards showed that only two beginning with these two letters were not for the prefix. The increase in processing time involved in adding another initial letter to the list of those looked for did not seem to be justified.

In the program employing the macro for production of an index to names in the MARC Pilot Project data, whenever the "doubt" signal was set, the name was printed on an edit list for human inspection. The name was then reprocessed with the "alternate" signal set and if a different output form was developed, this form also was printed. If the person reviewing the list accepted the first form, no special action was necessary. If the second was preferred, a card with an identifying number and the code 2 was punched; if a hand-made form was needed, this form was entered on a card with the code 3. These cards and the original output tape were then used to produce an edited output tape, in which the alternate forms were dropped unless a card directed otherwise. A second printed listing, recording the action taken, was also produced.

The doubtful cases identified by the algorithm are not limited to the prefix problems described above. By far the commonest occasion for doubt was the presence of "ä," "ö," or "ü." Was it a Germanic umlaut, calling for translation for filing purposes to "ae," "oe," or "ue," or was it something else? This is not the place to debate the practice, followed in most American academic libraries, of filing umlauted letters as if spelled out with an "e." The major bibliographies covering the German book trade do so, but most German dictionaries and encyclopedias do not; the example of other reference works and indexes is mixed. Since the aim of the work described here was to produce an index of names that could be used comfortably by librarians used to the practice, a means of continuing it was sought. However, it would be manifestly improper to insert an "e" if the mark were a diaeresis rather than an umlaut; and, in the opinion of the writer, almost equally improper for Hungarian, Finnish, and Turkish vowels. Even

those who do file such vowels in these languages as if they were Germanic do not usually do so for Chinese. It should be noted here that not all transformations of special letters turn on the doubt signal. "Å" is routinely translated to "aa" and Icelandic thorn to "th."

Other occasions for signalling doubt include names with a suspiciously high number of words before the first comma. This provision was introduced in an attempt to catch some non-names in the original data which had been wrongly coded, e.g. Women's Association of the St. Louis Symphony. When found, a card with the code D was punched for the edit run to delete these entries entirely.

Statistics of processing for the entire corpus of MARC Pilot Project data as cumulated and to some slight degree edited at the Harvard University Library will be useful in seeing the edit list in proper perspective. The entire file consisted of 47,884 records, 4,285 of which lacked names. The remaining 43,599 records contained 55,286 names (or alleged names). Of these, 52,372 or 94.7% were judged to be purely routine. Special processing of some sort not involving doubt (*e.g.*, recognition of compound surname, expansion of "Mc" to "Mac," closing up of apostrophe or non-doubtful prefix) was performed on 2,283 names, or 4.1%. The total number of doubtful names printed on the edit list was 631, or 1.1%. Somewhat more than half of these (334) resulted in different forms on being reprocessed with the "alternate" signal on. In 562 of the 631 doubtful cases, or 89% of this group, the first or only form printed was accepted, so that no action beyond inspection was necessary. Only 69 names, or not quite one out of 800 of the whole number, required the punching of a card—47 to indicate choice of the second form, 14 supplying a hand-made form, and 8 calling for deletion of non-names. Subsequent changes in the macro would have reduced considerably the number of names requiring hand-made forms.

It will be instructive to examine some of the names from the edit list to see what types of problems arise. The first selection of actual consecutive names (from LC card number 66-15363 through 66-17297) is rather typical:

Barnard, Douglas St. Paul		barnard 6douglas 7saint 7paul 3
Ekel⁴of, Gunnar,= 1907-		ekeloef 6gunnar 51907 3
	or:	ekelof 6gunnar 51907 3
Woolley, Al E.		woolley 6al 7e 3
Sch⁴onfeld, Walther H. P.,=1888-		schoenfeld 6walther 7h 7p 51888 3
	or:	schonfeld 6walther 7h 7p 51888 3
J⁴anner, Michael		jaenner 6michael 3
	or:	janner 6michael 3
M⁴uller, Alois,= 1924-		mueller 6alois 51924 3
	or:	muller 6alois 51924 3
Huang, Y⁴uan-shan		huang 6yuean 7shan 3
	or:	huang 6yuan 7shan 3
M⁴uller, Kurt,= 1903		mueller 6kurt 51903 3
	or:	muller 6kurt 51903 3

Note the dominance of simple umlauts; also, as a curiosity, the fact that all persons named "Al" appear on the list because of the possibility that it might be an unhyphenated Arabic prefix. Note also that Saint is treated as a separate word, not closed up as a prefix. "St." was originally put on the doubtful list with the thought that it might stand for Sankt or Szent instead of Saint, although normal library practice would not use an abbreviation in such cases. Its inclusion on the doubtful list was unexpectedly justified, however, by the occurrence of the name Erlich, Vera St. It seems likely that in this case "St." may stand for a patronymic, perhaps Stojanova or Stefanova, and there may be other occasions on which St. rather than S. is used as an abbreviation for such a name as Stefan (cf. the French use of Ch. rather than simple C. as an abbreviation for Charles).

The only action required for the names in the list above would be to punch a "2" card for the Chinese name Huang, Yüan-shan. Indeed, just as the umlaut is the largest category on the edit list, so the non-umlaut—a diacritic that looks like an umlaut but does not call for insertion of "e"—is the commonest occasion for punching an exception card. Occasionally a diaeresis is found:

Lecomte du No¹uy, Pierre		lecomte 7du 7nouey 6pierre 3
	or:	lecomte 7du 7nouy 6pierre 3

More common are certain front vowels in Hungarian, Finnish, or Turkish, or the vowel ü in Chinese as already encountered:

F¹oldi, Mih²aly		foeldi 6mihaly 3
	or:	foldi 6mihaly 3
T¹olgyessy, Juraj		toelgyessy 6juraj 3
	or:	tolgyessy 6juraj 3
Mett⁴al¹a-Portin, Raija		mettaelae 7portin 6raija 3
	or:	mettala 7portin 6raija 3
N¹arv¹anen, Sakari		naervaenen 6sakari 3
	or:	narvanen 6sakari 3
In⁴on⁴u, E.		inoenue 6e 3
	or:	inonu 6e 3
S¹umer, Mine		suemer 6mine 3
	or:	sumer 6mine 3
Y¹u, Ying-shih		yue 6ying 7shih 3
	or:	yu 6ying 7shih 3

Some libraries avoid the problem by treating all but the last of these as if umlauted, but determination of the correct category can usually be made at sight. Occasionally a name gives pause, for example these two which both prove to be Swiss and presumably Germanic, although Chönz may be Romansh:

Ch¹onz, Selina		choenz 6selina 3
	or:	chonz 6selina 3
R¹uede, Thomas		rueede 6thomas 3
	or:	ruede 6thomas 3

Somewhat more troublesome are names where some but not all elements are Germanic:

Vogt, Ulya (G4oknil)		vogt 6ulya 7goeknil 3
	or:	vogt 6ulya 7goknil 3
Ouchterlony, 4Orjan		ouchterlony 6oerjan 3
	or:	ouchterlony 6orjan 3
Iv2anyi-Gr4unwald, B2ela		ivanyi 7gruenwald 6bela 3
	or:	ivanyi 7grunwald 6bela 3

Although Vogt is obviously Germanic, Ulya Göknil is equally obviously not, and therefore the decision is that no umlaut is present. Örjan, on the other hand, is a Scandanavian forename, to be treated as umlauted even though coupled with a surname of Scottish Gaelic origin. Béla Iványi-Grünwald is a more difficult case. Grünwald is of course Germanic in origin, but can it be regarded as Magyarized? In English we might assume that such a name is Anglicized when the bearer starts writing it Grunwald or Gruenwald. However, the case is not so clear in Hungarian, since that language also has the letter "ü." Discussion of such a point may seem to split hairs, but it does involve a significant difference between manual and machine systems. In a manual system, the question of whether to file as Iványi-Grunwald or as Iványi-Gruenwald would arise only in the exceedingly unlikely event that another name which would file between the two also occurred in the corpus. In a machine system, however, any difference, even this late in a distinctive name, could result in the various works of the author being misfiled among themselves, or a work about him filed before one by him.

Use of different codes to represent the same graphic, umlaut on the one hand or diaeresis or other non-umlaut on the other, would drastically reduce both the number of doubtful names and the number of those for which an exception procedure is required. The Harvard College Library actually follows this practice. The Library of Congress experimented with it, but found that catalogers were reluctant in some cases to make the decision. Contemplation of the case of Béla Iványi-Grünwald gives the author more sympathy with this reluctance than he originally felt.

In attempting to evaluate the method described above, one must acknowledge both strong points and limitations. On the one hand it is very gratifying to see AEsopus and [Aesopus] falling together despite differences in the capitalization of the "e" and the bracketing, and to find such sequences as the following, all without even being referred to the edit list under the rules then prevailing:

Aziz, Khursheed Kamal	aziz 6khursheed 7kamal 3
Aziz Ahmad	aziz 7ahmad 3
al-Azm, Sadik J.	azm 6sadik 7j 3
Azrael, Jeremy R.	azrael 6jeremy 7r 3
Ba Maw, U	ba 7maw 6u 3
Baab, Clarence Theodore	baab 6clarence 7theodore 3

Delgado, David J.	delgado 6david 7j 3
Del Grande, John Joseph	delgrande 6john 7joseph 3
Delhom, Louis A.	delhom 6louis 7a 3
Delieb, Eric	delieb 6eric 3
DeLise, Knoxie C.	delise 6knoxie 7c 3
De Lisser, R. Lionel	delisser 6r 7lionel 3
Dell, Ralph Bishop	dell 6ralph 7bishop 3
Dellinger, Dave	dellinger 6dave 3
Dell'Isola, Frank	dellisola 6frank 3
Del Mar, Alexander	delmar 6alexander 3
Delmar, Anton	delmar 6anton 3
Delmar-Morgan, Edward Locker	delmar 7morgan 6edward 7locker 3

While it is certainly true that the system cannot survive without some provision for referring doubtful questions to a human editor, the number of these depends to a considerable extent on the filing and coding policies followed. Provided forename entries are coded as such, the system does a good job of identifying possible problems. (Presently, all multiple word forename entries are considered doubtful.) "Ua" has already been cited as an example of a prefix deliberately omitted, and there are others which could be added at any time it is thought worth while. A more troublesome situation, pointed out by Kelley Cartwright, is the possible occurrence of "Van" as a non-final element of an unhyphenated Vietnamese name. The only way this could be prevented from misfiling by merging it with the next element would be to throw all "Vans" including the numerous ones of Dutch origin into the doubtful category, expanding the edit list more than twenty percent. This did not seem advisable, particularly since normal library usage is to hyphenate Vietnamese compound names.

Up to this point the evaluation is quite favorable. The system can correctly process a very large proportion of names, including some which involve quite sophisticated points, without reference to a human editor, and it can call virtually all the rest to the attention of an editor. However, human review of problems means that there will be occasions when borderline cases are decided in different ways. If a permanent machine file of all established forms of names in the system is kept, both forms of each doubtful name could be checked against it so that decisions already made would not have to be repeated, thus saving the time of the editor as well as the hazard of differing decisions. It would of course be very expensive to keep such a file just for this purpose, but a file of this type would probably form a part of a comprehensive mechanized bibliographic system anyway.

Another area in which a mixed report would have to be given to the system is its extensibility to types of headings other than names. In work conducted on the same principles with a few thousand early titles from the MARC Pilot Project, there were only two conspicuous problems, one of which may not in fact be a problem: the filing of numbers as such rather

than as if they were spelled out in the language of the title. True, the particular algorithm then in use did not provide for bringing numbers of differing length into logical order ("50 great ghost stories" before "200 years of watercolor painting in America"), but this is a readily attainable refinement. The other problem is more refractory and is exemplified by titles beginning with prefix names, for example "De Gaulle," "De Soto," and "Van Gogh." Names within titles could not receive the usual name treatment since there was no way of identifying them as such, and therefore the prefixes were filed as separate words. Furthermore, while MARC Pilot Project authors were quite a cosmopolitan lot, the titles were almost entirely in English. Therefore, removal of initial articles was not much of a problem. There did not happen to be any work beginning "A to Z of . . .". However, there was a book which, although in English and so coded, had a title beginning with a Spanish article: "La vida," by the late Oscar Lewis. In working toward automatic removal of initial articles from titles, the usual assumption is that machine coding of the language of the work is available and will be checked first. This seems desirable both because it is probably more efficient in machine time than to check every title against a long list of possible articles in many languages, and because words that are articles in one language are not necessarily so in another. Most occurrences of initial "die" are probably German articles, but some are other parts of speech in English, for example "Die Casting" or "Die like a Dog."

If the umlaut is the common problem in names, the initial indefinite article which is the same as the numeral "one" in several languages may well be the most frequent occasion for doubt in processing of titles. "Un" or "ein" will usually mean "A," to be dropped; but will sometimes mean "One," to be kept. There are certainly other problems, in addition to the one with prefix names already mentioned, including some that give trouble even in manual filing: "Charles the First," "Charles II," "Charles V et son temps." It may be that at some point in the cataloging process a reviser will have to be on the lookout for certain of these special situations and add flags to indicate that a title includes a prefix name, or that it begins with an article which would not be found by program, or that it does not begin with an article although it appears to do so, or that for some other reason it calls for a hand made key.

The system described is not an absolute system, but absolute systems have their own tyrannies. If, as the author believes, Cartwright and Shoffner (2) are correct in thinking that a mixture of methods will be required in actual book catalog projects, then a system along the lines of the one described may well be a useful part of the mix.

REFERENCES

1. Nugent, William R.: "The Mechanization of the Filing Rules for the Dictionary Catalogs of the Library of Congress," *Library Resources & Technical Services,* 11 (Spring 1967), 145-166.

2. Cartwright, Kelley L.; Shoffner, Ralph M.: *Catalogs in Book Form* ([Berkeley]; Institute of Library Research, University of California, 1967), pp. 24-27.
3. Cartwright, Kelley L.: "Mechanization and Library Filing Rules," *Advances in Librarianship,* 1 (1970), 59-94.
4. Hines, Theodore C.; Harris, Jessica L.: *Computer Filing of Index, Bibliographic, and Catalog Entries* (Newark, N.J.: Bro-Dart Foundation, [1966]).
5. Harris, Jessica L.; Hines, Theodore C.: "The Mechanization of the Filing Rules for Library Catalogs: Dictionary or Divided," *Library Resources & Technical Services,* 14 (Fall 1970), 502-516.
6. Palmer, Foster M.: *A Macro Instruction to Process Personal Names for Filing* ([Cambridge, Mass.]: Harvard University Library, 1970). A copy of this document, which contains an Autocoder listing of the actual working macro, has been deposited with the National Auxiliary Publications Service, from which it can be obtained on microfiche (NAPS 01680). In this version there are only three codes, 2 corresponding to 3 as used in this paper, 4 to both 5 and 6, and 6 to 7. There are also a few differences in the treatment of particular prefixes. The macro is made up of 579 cards, of which 125 are comments only.